PCT/GB 00 / 0 0 4 9 5
15 FEBRUARY 2000

The Patent Office

INVESTOR IN PEOPLE

GB 00/495
ETU

PRIORITY DOCUMENT

SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

09/913458

I, the undersigned, being an officer duly authorised in accordance with Section ~~74(1) and (4)~~ of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

Signed Duncan

Dated 19 NOV 1999

Patents Form 1/77

Patents Act 1977
(Rule 16)

The Patent Office

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

The Patent Office

Cardiff Road
Newport
Gwent NP9 1RH

1. Your reference

30990085 GB

2. Patent application number
(The Patent Office will fill in this part)

9922663.1

3. Full name, address and postcode of the or of each applicant *(underline all surnames)*

Hewlett-Packard Company
3000 Hanover Street
Palo Alto
California 94304 USA

Patents ADP number *(if you know it)*

06293385001

If the applicant is a corporate body, give the country/state of its incorporation

Delaware USA

4. Title of the invention

A Method of Protecting the Confirugation of Modules in Computing Apparatus

5. Name of your agent *(if you have one)*

LAWMAN Matthew John Mitchell

"Address for service" in the United Kingdom to which all correspondence should be sent *(including the postcode)*

Hewlett-Packard Limited
IP Section
Filton Road
Stoke Gifford
BRISTOL BS34 8QZ

THE PATENT OFFICE
25 SEP 1999
RECEIVED BY POST

Patents ADP number *(if you know it)*

07337009001

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and *(if you know it)* the or each application number

Country	Priority application number *(if you know it)*	Date of filing *(day / month / year)*

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application	Date of filing *(day / month / year)*

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? *(Answer 'Yes' if:*

Yes

a) any applicant named in part 3 is not an inventor, or
b) there is an inventor who is not named as an applicant, or
c) any named applicant is a corporate body.
See note (d))

Patents Form 1/77

9. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form	
Description	13
Claim(s)	1
Abstract	
Drawing(s)	6 + 6

10. If you are also filing any of the following, state how many against each item.

Priority documents	
Translations of priority documents	
Statement of inventorship and right to grant of a patent *(Patents Form 7/77)*	
Request for preliminary examination and search *(Patents Form 9/77)*	X
Request for substantive examination *(Patents Form 10/77)*	
Any other documents *(please specify)*	

11. I/We request the grant of a patent on the basis of this application.

Signature M. Lawman Date 24/09/99

Matthew Lawman

12. Name and daytime telephone number of person to contact in the United Kingdom Katerina Nommeots-Nomm 0117 312 9947

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

a) If you need help to fill in this form or you have any questions, please contact the Patent Office on 0645 500505.

b) Write your answers in capital letters using black ink or you may type them.

c) If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.

d) If you have answered 'Yes' Patents Form 7/77 will need to be filed.

e) Once you have filled in the form you must remember to sign and date it.

f) For details of the fee and ways to pay please contact the Patent Office.

A Method of Protecting the Configuration of Modules in Computing Apparatus

[30990085]

This invention relates to a trusted device, a portable security token, secure communications between the device and token, and a method and development based on the secure communications, of protecting the configuration of modules in computing apparatus.

In today's computing world, most of all modules used in computing apparatus are standardised and exchangeable for anybody. It is good for making computing apparatus with low prices and with plenty of replacements of worn-out modules, but this results in a potential problem that it is not difficult for a person to reassemble a forged apparatus by using retailed or stolen modules.

How to make stealing computing apparatus less interesting has attracted much research attention. There are various existing proposals to render stolen apparatus inoperable and thus to prevent use by the thief. For instance, some apparatus has a security device, e.g. an immobiliser, a disabling unit or an application specific integrated circuit, that is used to control continued operation of the apparatus. This security device has secure communications with a remote control station. Allowing continued operation of the apparatus is dependent on a specific instruction signal from the remote control station. If the owner of the apparatus notifies the remote station that the apparatus has been stolen, the remote station will ensure that the appropriate instruction is not transmitted in the next validation routine and so the apparatus will not function properly thereafter. Also, if for any reason the apparatus is not in communication with the remote station during a validation routine, it will not function properly.

It is obvious that the above solution can help to prevent the thief from use of the stolen apparatus. However, some stealers are more interested in reassembling a forged apparatus by using stolen modules than in using the stolen apparatus directly, because they can get more benefit from this "business". In this case, the above solution can not help to reduce their interest of stealing apparatus. So a more common and harder problem is how to protect configuration of modules in computing apparatus and thus to prevent use of stolen modules by the thief.

For the purpose of the present invention, a module must have the property that its existence and functions can be verified by the host platform comprised of the set of modules. Typically, the identity of a module is untrustworthy for an ordinary apparatus because the ordinary apparatus is not able to verify the correctness of the module's "identity". For this reason, the present invention is not designed to sort out

the problem that stops the thief using stolen modules to reassemble any forged (or untrustworthy) apparatus. This invention is focused on how to prevent the thief from reassembling a "trusted platform" using stolen modules. We believe it will help to reduce interest in stealing modules of computing apparatus, in particular, when a trusted platform becomes a standardised model, any untrustworthy platform can be checked and it will be avoid by computer clients who want their computing apparatus to be trusted.

In order to protect configuration of modules in computing apparatus, the present inventors propose a new arrangement to establish a more trusted relationship amongst a host platform, a portable security token and a group of modules used in the host platform. Typically, the arrangement implements a security control policy to establish a module configuration profile that lists the registered module group, and to authenticate the modules listed with help of the portable security token.

Preferred embodiments of the invention implement mutual/unilateral authentication and privilege restriction. In particular, preferred embodiments utilise a novel method of binding the identity of the portable security token with varieties of the modules.

In accordance with a first aspect, the present invention provides computing apparatus comprising:

memory means storing the instructions of a secure process and an authentication process;

processing means arranged to control the operation of the computing apparatus including by executing the secure process and the authentication process as required;

user interface means arranged to receive user input and return to the user information generated by the processing means in response to the user input;

interface between the computing apparatus and a portable security token means for receiving the token and communicating with the token, the token comprising a body supporting:

a token interface for communicating with the interface means;

a token processor; and

token memory storing token data including information for identifying the token;

wherein the processing means is arranged to receive the identity information from the portable token, authenticate the token using the authentication process and, if the token is successfully authenticated, permit a user to interact with

the secure process via the user interface means for the purpose of establishing and modifying a module configuration profile comprising

a list of registered modules;

type, model, identity and other related information of each module included in the list; and

if it is not possible to authenticate the portable token, suspending the interaction between the computing apparatus and the user.

In accordance with a second aspect, the present invention provides a method of controlling computing apparatus to authenticate a module listed in the module configuration profile via

an interface between a trusted component and the module means for the trusted component receiving the module and communicating with the module, the module comprising a body supporting:

a module interface for communicating with the interface means;

module memory storing memory data including information for identifying the module; and

the trusted component comprising a body supporting:

a component interface for communicating with the above interface means;

a component processor; and

component memory storing component data including information for identifying the component,

wherein the processing means is arranged to receive the identity information from the modules, authenticate the module using the authentication process and, if the module is successfully authenticated, permit a user to interact with the secure process via the user interface means, and

if it is not possible to authenticate the module, suspending the interaction between the computing apparatus and the user.

Other aspects of the invention will become apparent from the accompanying description, claims and drawings.

Preferred embodiments of the present invention will now be described, by way of example only, with reference to the accompanying drawings, of which:

Figure 1 is a diagram that illustrates a system capable of implementing embodiments of the present invention;

Figure 2 is a diagram which illustrates a motherboard including a trusted device arranged to communicate with a Module Configuration Authority (MCA) smart card via a smart card reader and with a group of modules;

Figure 3 is a diagram that illustrates the trusted device in more detail;

Figure 4 is a diagram that illustrates the operational parts of a MCA smart card according to the present invention;

Figure 5 is a flow diagram that illustrates the process of mutually authenticating a MCA smart card and a host platform;

Figure 6 is a flow diagram that illustrates one example for a host platform to authenticate a module with cryptographic identity;

Figure 7 is a flow diagram that illustrates one example for the host platform to authenticate a module with serial number identity;

Figure 8 is a flow diagram that illustrates one example for the host platform to verify the authorisation of a module without distinguishable identity.

While the ideas of the invention are general, for ease of discussion, we will focus on preferred embodiments, wherein a smart card, is the Module Configuration Authority (MCA) security token, or say "MCA smart card", which interacts with a trusted computing platform, or simply "platform".

The present invention is proposed to prevent an unauthorised person from reconfiguring the platform. For this purpose, each apparatus has a module configuration profile. The profile includes a list of registered modules. The attributes of each module listed in the profile may include type, model, manufacturer, statistically unique identity if there exists one, usage privilege and other related information. In preferred embodiment of the present invention, this module configuration profile is stored in a portable security token, such as the MCA smart card, which has secure communications with a trusted device of the trusted computing platform.

The trusted device and the security token could be a secure pair with a trusted relationship based on strong authentication between each other.

By transferring the module configuration profile to the trusted device, the security token introduces each module listed in the profile to the trusted device, which is then able to authenticate those modules. It is required that both the security token and the trusted device have a function of tamper-resistant storage to store the module configuration profile.

After the trusted computing platform is reset, the trusted device checks whether every module present in the platform is listed in the module configuration profile. For the sake of simplicity of description, only three types of modules are considered in any detail herein:

- A module has a cryptographic identity that is of at least one private key for some cryptographic functions, such as signature and/or decryption. This

module can be authenticated by the trusted device without on-line help of the MCA smart card. Examples of this type of modules include a smart card with cryptographic functions, a cryptographic coprocessor, a drive hard disk with security functions and etc.;

- A module has a built-in serial number as an identity, which is statistically unique and is stored inside the module in a tamper-resistant fashion. This module may or may not be able to be authenticated by the trusted device without on-line help of the MCA smart card. Examples of this type of module include a smart card with secure storage function, a network card, Intel Pentium III and etc.;
- A module has no distinguishable identity. This module is exchangeable for anybody and the authorisation of usage of the module can be ensured with on-line help of the MCA smart card. For example, when the trusted device meets a module without a distinguishable identity, the device will ask for presentation of the MCA smart card to confirm a valid authorisation of the module.

A trusted platform 10 incorporating a smart card reader 12 is illustrated in the diagram in Figure 1. The platform 10 includes the standard features of a keyboard 14, mouse 16 and visual display unit (VDU) 18, which provide the physical 'user interface' of the platform. Along side the smart card reader 12, there is illustrated a smart card 19, say Module Configuration Authority (MCA) smart card. In the platform 10, there are plural modules 15, which are listed in the module configuration profile stored in the MCA smart card 19.

For convenience, the present invention uses the platform that is described in detail in the present applicant's co-pending European patent application EP 99301100.6 entitled "Trusted Computing Platform" filed on 15 February 1999. While this platform is ideal for the purposes of the present embodiment, and allows a user to verify the integrity of the platform, use of the platform is by no means essential to the present invention. The main features of a platform including a trusted device will now be reproduced herein for the reader's convenience.

As illustrated in Figure 2, the motherboard 20 of the trusted computing platform 10 (as described in the applicant's co-pending application) includes (among other standard components) a main processor 21, main memory 22, a trusted device 24, a data bus 26 and respective control lines 27 and lines 28, BIOS memory 29 containing the BIOS program for the platform 10 and an Input/Output (IO) device 23, which controls interaction between the components of the motherboard and the

smart card reader 12, the keyboard 14, the mouse 16 and the VDU 18. The main memory 22 is typically random access memory (RAM). In operation, the platform 10 loads the operating system, for example Windows NT™, into RAM from hard disk (not shown). Additionally, in operation, the platform 10 loads the processes or applications that may be executed by the platform 10 into RAM from hard disk (not shown). In the present case, the key applications are an authentication application and a secure application, the operations of which will be described below.

Typically, for a conventional IBM-compatible platform (i.e. a "PC"), the BIOS program is located in a special reserved memory area, the upper 64K of the first megabyte do the system memory (addresses FØØØh to FFFFh), and the main processor is arranged to look at this memory location first, in accordance with an industry wide BIOS standard.

A significant difference between the present trusted platform 10 and a conventional platform is that, after reset, the main processor 21 is initially controlled by the trusted device 24, which then hands control over to the platform-specific BIOS program, which in turn initialises all input/output devices as normal. After the BIOS program has executed, control is handed over as normal by the BIOS program to the operating system.

The trusted device 24 comprises a number of blocks, as illustrated in Figure 3. After system reset, the trusted device 24 performs a secure boot process to ensure that the operating system of the platform 10 (including the system clock and the display on the monitor) is running properly and in a secure manner. During the secure boot process, the trusted device 24 acquires an integrity metric of the computing platform 10. The trusted device 24 can also perform secure data transfer and, for example, authentication between it and a smart card via encryption/decryption and signature/verification. The trusted device 24 can also securely enforce various security control policies to be discussed below including verification of module configuration and the locking of user interface.

Specifically, the trusted device comprises: a controller 30 programmed to control the overall operation of the trusted device 24, and interact with the other functions on the trusted device 24 and with the other devices on the motherboard 20; a measurement function 31 for acquiring the integrity metric from the platform 10; a cryptographic function 32 for signing, encrypting or decrypting specified data; an authentication function 33 for authenticating a smart card; and interface circuitry 34 having appropriate ports (36, 37 & 38) for connecting the trusted device 24 respectively to the data bus 26, control lines 27 and address lines 28 of the motherboard 20. Each of the blocks in the trusted device 24 has access (typically via

the controller 20) to appropriate volatile memory areas 37 and/or non-volatile memory areas 38 of the trusted device 24. Additionally, the trusted device 24 is designed, in a known manner, to be tamper resistant.

For reasons of performance, the trusted device 24 may be implemented as an application specific integrated circuit (ASIC). However, for flexibility, the trusted device 24 is preferably an appropriately programmed micro-controller. Both ASICs and micro-controllers are well known in the art of microelectronics and will not be considered herein in any further detail.

One item of data stored in the non-volatile memory of the trusted device 24 is a certificate 350. The certificate 350 contains at least a public key 351 of the trusted device 24 and an authenticated value 352 of the platform integrity metric measured by a trusted party (TP). The certificate 350 is signed by the TP using the TP's private key prior to it being stored in the trusted device 24. In later communications sessions, a user of the platform 10 can verify the integrity of the platform 10 by comparing the acquired integrity metric with the authentic integrity metric 352. If there is a match, the user can be confident that the platform 10 has not been subverted. Knowledge of the TP's generally-available public key enables simple verification of the certificate 350. The non-volatile memory 35 also contains an identity (ID) label 353. The ID label 353 is a conventional ID label, for example a serial number, that is unique within some context. The ID label 353 is generally used for indexing and labelling of data relevant to the trusted device 24, but is insufficient in itself to prove the identity of the platform 10 under trusted conditions.

The trusted device 24 is equipped with at least one method of reliably measuring or acquiring the integrity metric of the computing platform 10 with which it is associated. In the present embodiment, the integrity metric is acquired by the measurement function 31 by generating a digest of the BIOS instructions in the BIOS memory. Such an acquired integrity metric, if verified as described above, gives a potential user of the platform 10 a high level of confidence that the platform 10 has not been subverted at a hardware, or BIOS program, level. Other known processes, for example virus checkers, will typically be in place to check that the operating system and application program code has not been subverted.

The measurement function 31 has access to: non-volatile memory 35 for storing a hash program 354 and a private key 355 of the trusted device 24, and volatile memory 42 for storing the public keys and associated ID labels 360a-360n of one or more authentic smart cards 19s that can be used to gain access to the platform 10 and an acquired integrity metric in the form of a digest 361.

A processing part 40 of a smart card 19 is illustrated in Figure 4. As shown, the smart card 19 40 has the standard features of a processor 41, memory 42 and interface contacts 43. The processor 41 is programmed for simple challenge/response operations involving authentication of the smart card 19 and verification of the platform 10, as will be described below. The memory 42 contains its private key 420, its public key 428, a module configuration profile 421, the public key 422 of the TP and an identity 427. The module configuration profile 421 lists the registered modules 15 AC1-ACn usable by the apparatus, and the individual security policy 424 for the apparatus. For each module 15, the module configuration profile includes respective identification information 423, the trust structure 425 between the modules (if one exists) and, optionally, the type or make 426 of the module.

In the module configuration profile 421, each module 15 entry AC1-ACn includes associated identification information 423, which varies in dependence upon the type of modules, as described above.

The 'security policy' 424 dictates the options that a user has on the platform 10 while verifying a module 15. For example, the user interface may be locked or unlocked while a module 15 is authenticated, depending on the function of the module 15. Additionally, or alternatively, certain files or executable programs on the platform 10 may be made accessible or not, depending on how trusted a particular module 15 is. Further, while authentication for a module 15 is failed, the user interface may be locked, that means there is no access available for the user, or alternatively, the user interface may be unlocked, where only the functions related with this module is not available for the user.

A 'trust structure' 425 defines whether a module 15 can itself 'introduce' further modules 15 into the system without first re-using the MCA smart card 19. In the embodiments described in detail herein, the only defined trust structure is between the MCA smart card 19 and the modules 15 that can be introduced to the platform 10 by the MCA smart card 19. This would require a module 15 to have an equivalent of a module configuration profile listing the or each module that it is able to introduce. To prevent misuse, such a module must be removable, and be stored apart from the host platform.

A preferred process for authentication between an MCA smart card 19 and a platform 10 will be described with reference to the flow diagram in Figure 5. As will be described, the process conveniently implements a challenge/response routine. There exist many available challenge/response mechanisms. The implementation of an authentication protocol used in the present embodiment is mutual authentication with 3-pass, as described in ISO/IEC 9798-3 [1]. Of course, there is no reason why other

authentication procedures cannot be used, for example 2-pass or 4-pass, as also described in [1].

Initially, the user inserts their MCA smart card 19 into the smart card reader 12 of the platform 10 in step 500. Then, the trusted device 24 is triggered to attempt mutual authentication by generating and transmitting a nonce A to the MCA smart card 19 in step 505. A nonce, such as a random number, is used to protect the originator from deception caused by replay of old but genuine responses (called a 'replay attack') by untrustworthy third parties.

In response, in step 510, the MCA smart card 19 generates and returns a response comprising the concatenation of: the plain text of the nonce A, a new nonce B generated by the MCA smart card 19, the ID 353 of the trusted device 24 and some redundancy; the signature of the plain text, generated by signing the plain text with the private key of the MCA smart card 19; and a certificate containing the ID and the public key of the MCA smart card 19.

The trusted device 24 authenticates the response by using the public key in the certificate to verify the signature of the plain text in step 515. If the response is not authentic, the process ends in step 520. If the response is authentic, in step 525 the trusted device 24 generates and sends a further response including the concatenation of: the plain text of the nonce A, the nonce B, the ID 427 of the MCA smart card 19 and some redundancy; the signature of the plain text, generated by signing the plain text using the private key of the trusted device 24; and the certificate comprising the public key of the trusted device 24 signed by the private key of the TP.

The MCA smart card 19 authenticates this response by using the public key of the TP in step 530. If the further response is not authentic, the process ends in step 535.

If the procedure is successful, both the trusted device 24 has authenticated the MCA smart card 19 and the MCA smart card 19 has verified the trusted device of the trusted platform 10 and, in step 540, the authentication process executes some secure processes between the trusted device 24 and the MCA smart card 19, for example, transferring the module configuration profile from the MCA smart card 19 to the trusted device 24.

Additionally, or alternatively, in some embodiments it may be required that the module configuration profile is encrypted and signed for transferring to protect privacy and integrity. If so, a secure data transfer protocol may be needed between the trusted device 24 and the MCA smart card 19. There exist many available mechanisms for transferring secure credentials between two entities. A possible

implementation, which may be used in the present embodiment, is secure key transport mechanisms from ISO/IEC DIS 11770-3 [2].

According to the requirement of different applications, the trusted device 19 and the MCA smart card may be a specific pair, and a user is not able to use one smart card to authorise more than one platform and/or a platform cannot be authorised by using more than one smart card. Otherwise, the trusted device 24 and the MCA smart cards 19 may have more complicated relationship, such as one smart card can work for more than one platform and/or one platform can be authorised by using more than one smart card.

There may be a recovery service for MCA smart cards. For example, if one smart card is lost, the owner of the corresponding platform can ask the service to change the authority relationship from the lost MCA smart card to some other MCA smart card. But this change must be very careful and communications with the trusted device 24 must be authenticated by a designated authority.

A preferred process for authenticating a cryptographic identity module 15 by a platform 10 will be described with reference to the flow diagram in Figure 6. As will be described, the process conveniently implements a challenge/response routine. Again, there exist many available challenge/response mechanisms. The implementation of an authentication protocol used in the present embodiment is unilateral authentication with 2-pass, as described in ISO/IEC 9798-3 [1]. Of course, there is no reason why other authentication procedures cannot be used, for example 1-pass, as also described in [1].

Initially in step 600, the trusted device 24 retrieves a module configuration profile listing the identity information of the module, which may be a certificate of a public key corresponding with the module's private key. It is assumed that the trusted device 24 can verify the validation of the certificate of the module's public key. It then challenges the module by sending a nonce in step 605. After receiving the nonce, in step 610, the MCA smart card 19 generates and returns a response comprising the concatenation of: the plain text of the nonce, the ID 353 of the trusted device 24 and some redundancy; the signature of the plain text, generated by signing the plain text with the private key of the MCA smart card 19; and a certificate containing the ID and the public key of the MCA smart card 19.

The trusted device 24 authenticates the response by using the public key in the certificate to verify the signature of the plain text in step 615. If the response is not authentic, the process ends in step 620. If the response is authentic, in step 630, the authentication process executes some secure processes between the trusted device 24 and the MCA smart card 19.

A preferred process for authenticating a serial number identity module 15 by a platform 10 will be described with reference to the flow diagram in Figure 7. The trusted device 24 needs to check if the serial number and other related information of the module match with the data about this module listed in the module configuration profile. According to the verification of integrity of the trusted platform 10, as described above, it is assumed that the communications between the security device 24 and the module 15 can not be tampered with. If this condition is not existing, the on-line help of the MCA smart card will be required, as described in verification of authorisation of a module without self-identity below.

Initially, the trusted device 24 retrieves a module configuration profile listing the identity information of the module in step 700, then it requests the serial number of the module in step 705. The module 15 returns the response with its serial number in step 710. The trusted device 24 compares this serial number with the data recorded in the module configuration profile in step 715. If it matches, the authentication passes and the following secure process will carry on in step 730; otherwise, the authentication is failed in step 720.

A preferred process for verifying authorisation of a module without self-identity 15 by a platform 10 will be described with reference to the flow diagram in Figure 8. The authorisation of usage of the module can be ensured with on-line help of the MCA smart card.

Initially, the trusted device 24 retrieves a module configuration profile listing the identity information of the module in step 800. When the trusted device 24 meets a module 15 without a distinguishable identity, the trusted device 24 will ask for presentation of the MCA smart card 19 to confirm a valid authorisation of the module. To do so, the trusted device first displays a message to request an MCA smart card 19 in step 825, and second locks the user interface in step 830. The user inserts the MCA smart card 19 in step 833. Authentication between the trusted device 24 and the MCA smart card 19 can choose either unilateral authentication or mutual authentication as shown above. In Figure 8, we use a unilateral authentication with 2-pass, as described in ISO/IEC 9798-3 [1]. The trusted device 24 challenges the MCA smart card 19 in step 840, and the MCA smart card 19 responses in step 845. The trusted device 24 authenticates the response in step 850. If the response is not authentic, the process aborts in step 855. If the response is authentic, the trusted device accepts the corresponding module, and the following secure process will carry on in step 860.

In the above cases of authentication process of modules, for any reason if the authentication process is aborted, the user interface may be temporarily locked

dependent on the security policy 424. On the other hand, if the user inserts the MCA smart card 19, a new round of authentication is performed between the MCA smart card and the platform 10. Upon successful verification, the user interface will be unlocked and the user is able to change the module profile to authorise the module.

It is obvious that in this method, the owner of the trusted computing platform 10 and the MCA smart card 19 is strongly recommended to maintain the MCA smart card 19 safely, in particular, maintaining it separately with the platform 10 as soon as it is not communicating with the platform. For the purpose of protecting the smart card more securely, it is recommended to use a password known to only the owner to access the MCA smart card 19.

This method can be extended to a remote control model, where the security token is not connected with the apparatus locally and is connected with the apparatus via a network, e.g., the Internet. For example, a security token, such as a smart card, can be located in another computing platform connected with the Internet. The location of the smart card is portable. In this case, the communication between the security token and the apparatus is through over the Internet.

Although, it is not necessary for the security token to be centrally controlled, and its location is portable, this solution may work with a remote control centre to make more powerful and more flexible prevention from usage of a stolen apparatus and stolen modules. We do not want to replace the portable security token with the remote security centre completely. We prefer to offer more than one option to the owner of apparatus. For instances, the owner may be able to choose either use of a portable token only or a remote control centre or both of them. If the owner chooses the latest, the trusted device inside apparatus must communicate correctly with both the remote control station and the portable security token, to let the apparatus be able to function properly and the modules of the apparatus be configured or reconfigured properly. With collaboration of the local control and the remote control, the owner can get more powerful service. They may notify the remote security station when the apparatus has been stolen with the portable token.

REFERENCES:

[1] ISO/IEC 9798-3, Second Edition, "Information technology – Security techniques – Entity authentication – Part 3: Mechanisms using digital signature techniques", International Organization for Standardization, 15 October 1998.

[2] ISO/IEC DIS 11770-3.2, "Information technology – Security techniques – Key management – Part 3: Mechanisms using asymmetric techniques", International Organization for Standardization, 29 July 1997.

Claims

1. A method of protecting the configuration of modules in computing apparatus, in which a trusted device is used for allowing operation of the modules listed in a module configuration profile, that is dependent on authentication of identities of the modules and verification of authority presentation from a portable security token.

2. Authentication process between the trusted device and the portable security token and security credential (in particular, the information of the module configuration profile) transfer from the portable security token to the trusted device based on communications between apparatus and the portable security token in accordance with claim 1.

3. Authentication process where the trusted device is able to authenticate the cryptographic identity of a module listed in the module configuration profile in accordance with claims 1 and 2.

4. Authentication process where the trusted device is able to authenticate a module with a built-in serial number identity listed in the module configuration profile, in accordance with claims 1 and 2.

5. Authorisation verification process where the trusted device is able to verify the authority of usage of a module without distinguishable identity listed in the module configuration profile, in accordance with claims 1 and 2.

6. A trusted computing platform with one or more physical trusted devices arranged for use in accordance with any one of claim 1 and claim 2.

7. Smart cards are used as portable security tokens in accordance with any one of claims 1 to 5.

8. Various polices, such as more than one token for one platform, or one token for more than one platform arranged for use in accordance with claim 1.

9. Remote control using either smart cards or centre services or both arranged for use in accordance with claim 1.

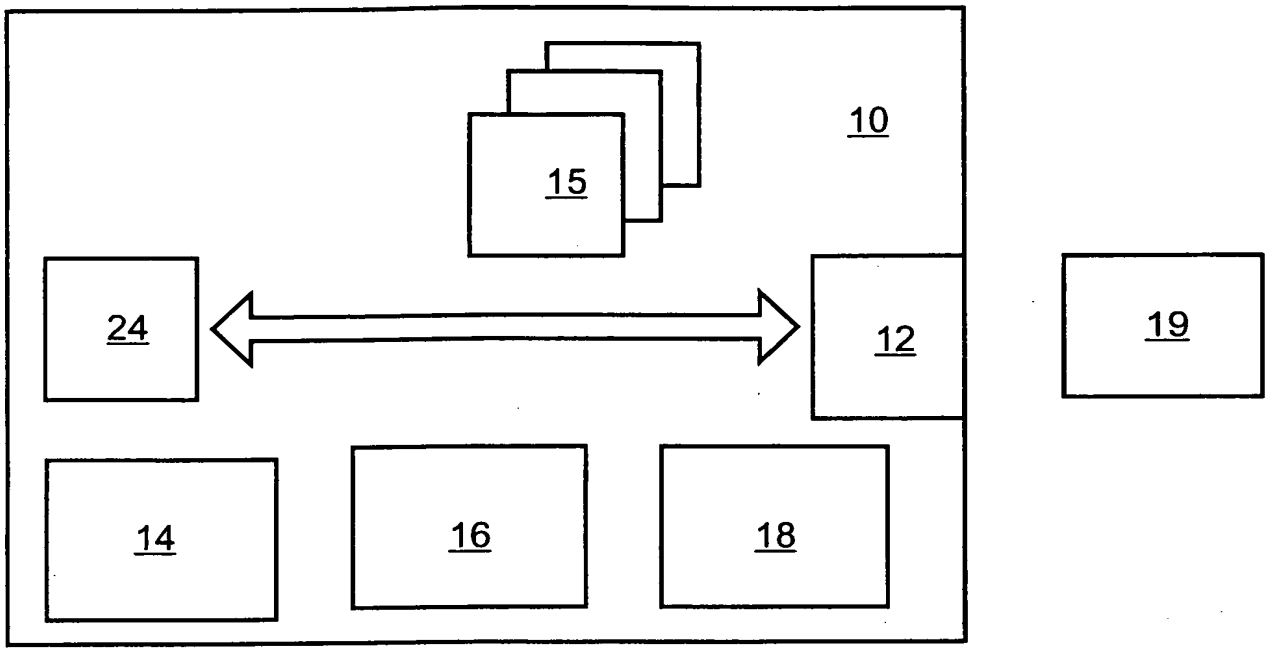

FIGURE 1

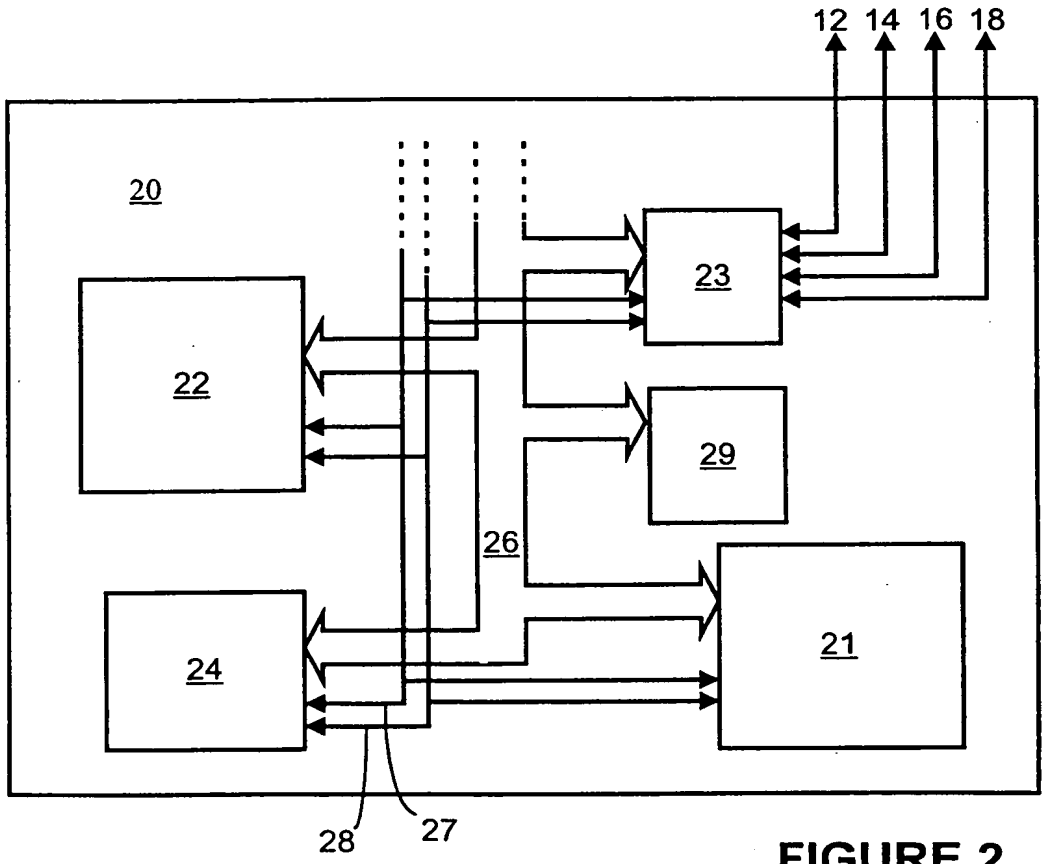

FIGURE 2

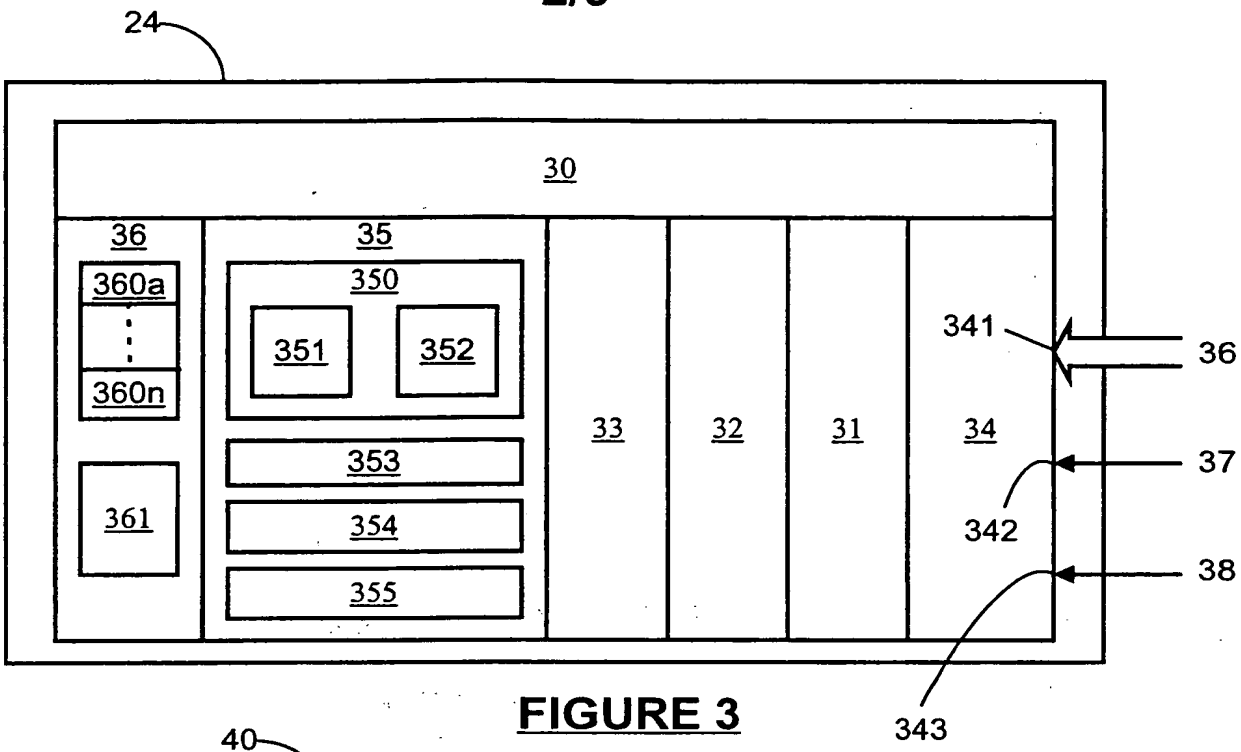

FIGURE 3

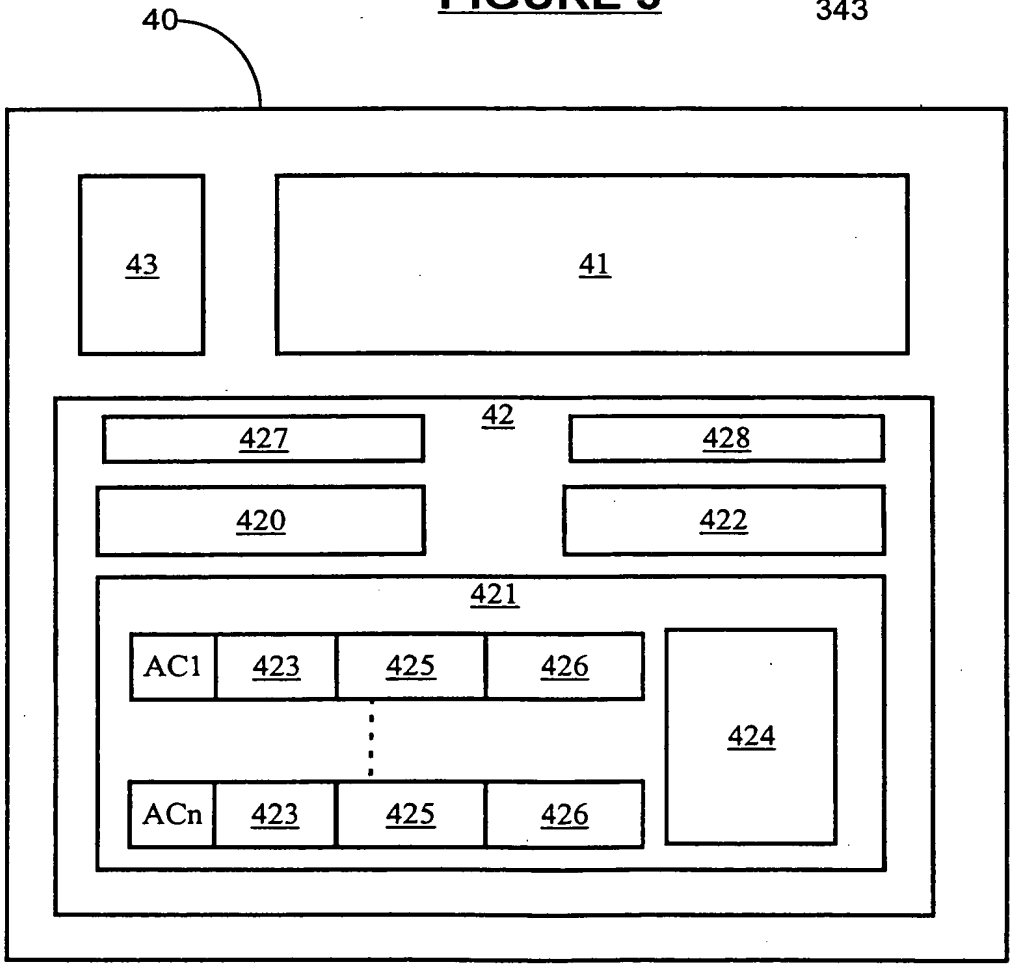

FIGURE 4

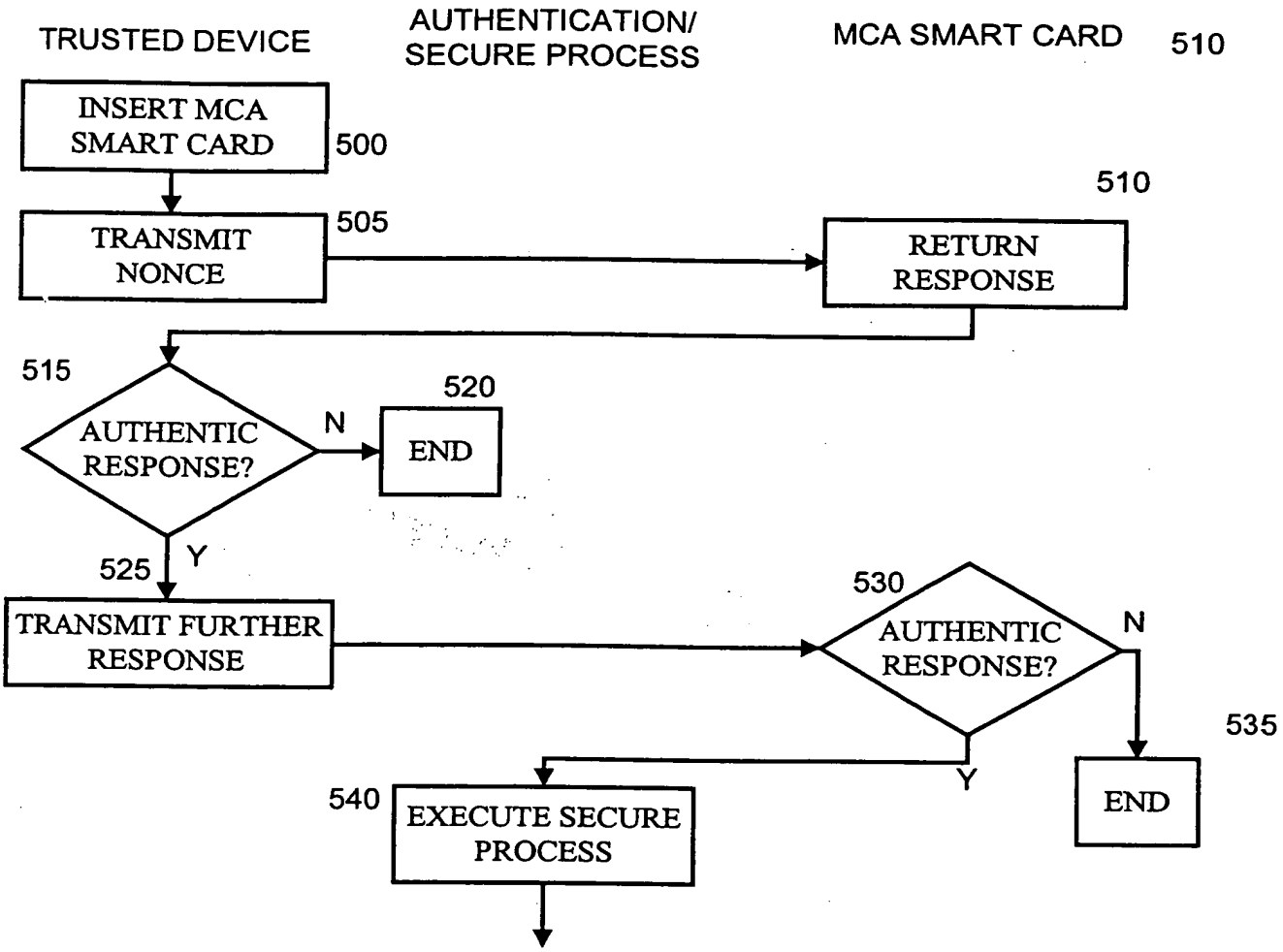
TRUSTED DEVICE
AUTHENTICATION/ SECURE PROCESS
MCA SMART CARD
510
INSERT MCA SMART CARD
500
TRANSMIT NONCE
505
RETURN RESPONSE
510
515
AUTHENTIC RESPONSE?
N
520
END
Y
525
TRANSMIT FURTHER RESPONSE
530
AUTHENTIC RESPONSE?
N
535
END
Y
540
EXECUTE SECURE PROCESS

FIGURE 5

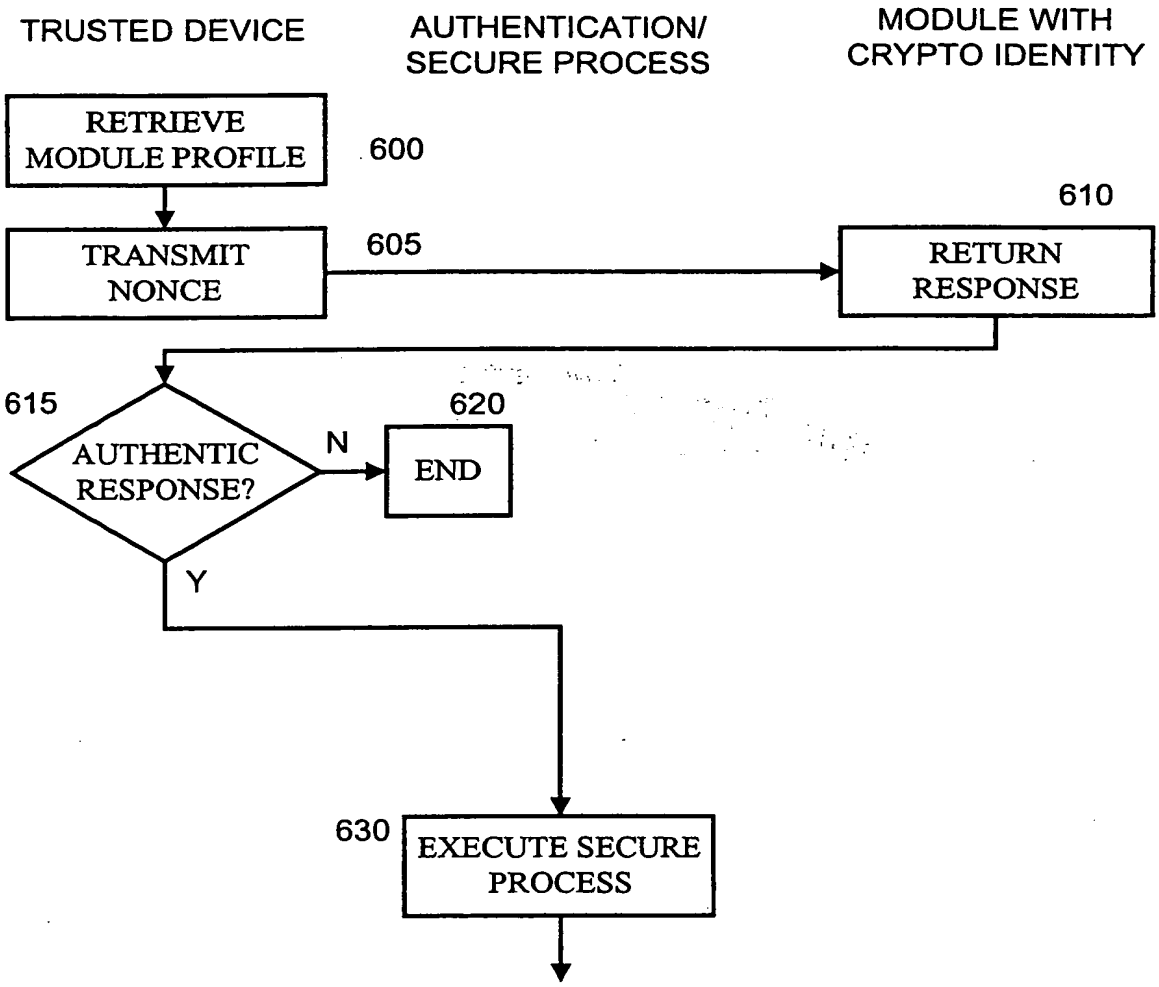
TRUSTED DEVICE
AUTHENTICATION/ SECURE PROCESS
MODULE WITH CRYPTO IDENTITY
RETRIEVE MODULE PROFILE
600
610
TRANSMIT NONCE
605
RETURN RESPONSE
615
620
AUTHENTIC RESPONSE?
N
END
Y
630
EXECUTE SECURE PROCESS

FIGURE 6

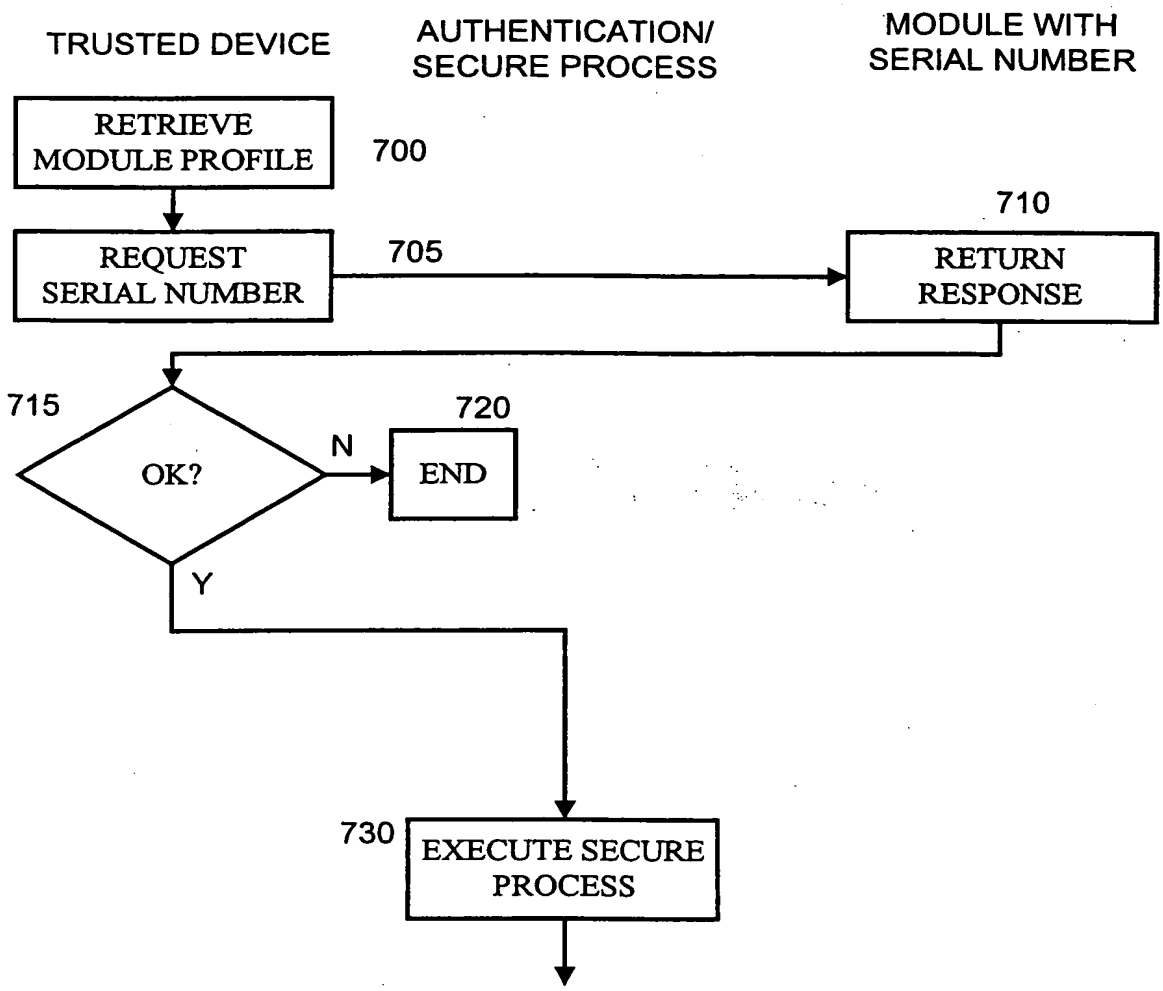
TRUSTED DEVICE
AUTHENTICATION/ SECURE PROCESS
MODULE WITH SERIAL NUMBER
RETRIEVE MODULE PROFILE
700
710
REQUEST SERIAL NUMBER
705
RETURN RESPONSE
715
720
OK?
N
END
Y
730
EXECUTE SECURE PROCESS

FIGURE 7

TRUSTED DEVICE

AUTHENTICATION/
SECURE PROCESS

MODULE WITHOUT
IDNETITY/
MCA SMART CARD

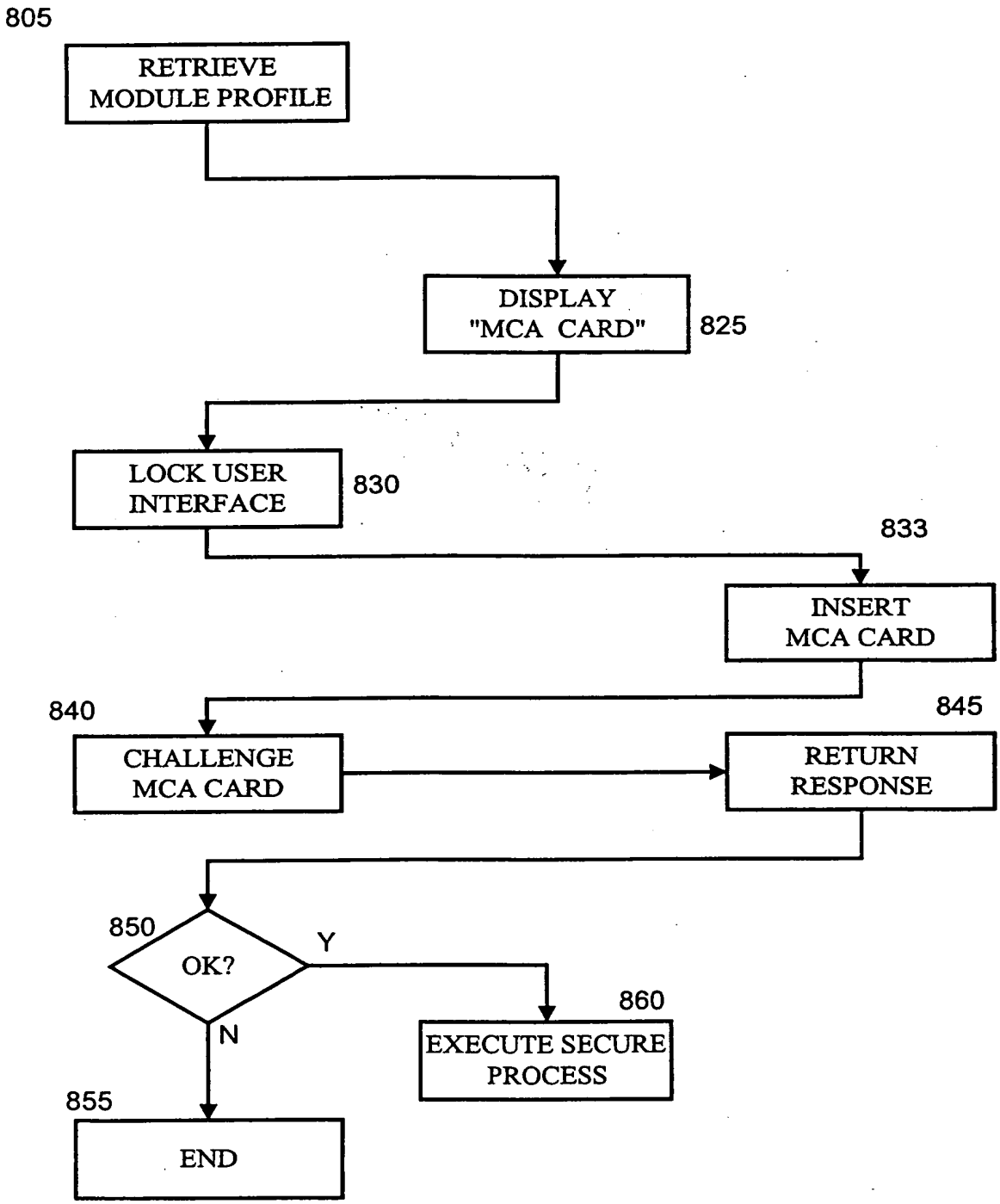

FIGURE 8